学術論文の章構造に基づくニューラル自動要約モデル

衣川 和尭　　鶴岡 慶雅
東京大学大学院 工学系研究科
{kinugawa, tsuruoka}@logos.t.u-tokyo.ac.jp

1 はじめに

文章自動要約とは入力された文章に対して，内容的に核となる部分を保持しつつ短くまとめた文章を生成する技術のことである．自然言語処理においては，入力文書内を文単位で区切って，その中で重要な文を数文抽出することで要約を生成する「抽出型」の要約と，人間と同様に本文をパラフレーズしたり再構成したりして一から文章を生成する「生成型」の要約に分けられるが,「抽出型」の要約は「生成型」の要約に比べて計算コストが少ないことや，意味・文法的に正しい要約を出力しやすいことから研究の主流となっている．

自動要約の研究対象の文書としてはニュースや新聞など様々なものがあるが，その一つとして学術論文が挙げられる．論文には元々アブストラクトが含まれており，それは本文に対する一つの質の高い参照要約とみなせるので評価が行いやすいことや，論文データベースから大量に用意することが可能であるので機械学習に適していることが理由であるが，論文に対する要約器を構築することで，extended abstract のように元々のアブストラクト以上の内容を含んだ要約を生成するといった応用も考えられる．一方で，論文は文書のサイズが比較的大きいため要約が難しくなるという問題もある．

そこで本研究では，論文の談話構造を活用した，教師あり学習に基づく要約器のモデルを構築することを目指す．一般に，文章が長いときは作者も読み手のために節や章を設けて，明示的に文書を構造化して情報を整理する傾向があり，このような談話構造を活用することは，単に要約の精度向上が見込めるだけでなく，作者の意図した論理展開を汲むことができるという意味で要約の本質をなしている．

本論文では，Encoder-Decoder 機構を自動要約タスクに適用したモデルをベースに，論文の章構造を活用したモデルを検討した．

2 関連研究

近年自動要約の分野では，機械翻訳の研究で盛んに用いられている Encoder-Decoder モデルを要約タスクに適用する研究が現れている．Cheng と Lapata は Encoder 側で入力文書の全文を読み込み，Decoder 側で各文が要約として抽出される妥当性のスコアを出力するモデルを提案した [2]．モデルの構造を図 1 に示す．このモデルでは Embedding された単語を基に文のベクトルを取得したあと，各文ベクトルを順次リカレントニューラルネットワーク (以下 RNN) に入力する．文書 (つまり文ベクトルの系列) $\boldsymbol{d} = [\boldsymbol{s}_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_m]$ に対して，t 番目の文ベクトル $\boldsymbol{s}_t$ を入力した際の RNN ユニットの出力 $\boldsymbol{h}_t$ は以下のように計算される．

$$\boldsymbol{h}_t = \mathrm{RNN}_{\mathrm{enc}}(\boldsymbol{s}_t, \boldsymbol{h}_{t-1}) \tag{1}$$

全文を読み終わったら，最後尾のユニットの出力を Decoder 側の RNN に接続する．Decoder 側でも同様に文ベクトルを読み込んでいくが，Encoder 側とは異なり，今度は要約器によって予測された要約文としての妥当性の程度 $p \in [0, 1]$ を掛け合わせて入力する．

$$\bar{\boldsymbol{h}}_t = \mathrm{RNN}_{\mathrm{dec}}(p_{t-1}\boldsymbol{s}_{t-1}, \bar{\boldsymbol{h}}_{t-1}) \tag{2}$$

Encoder 側では，各文を順次 RNN に読み込ませることで文章全体をエンコーディングしたが，Decoder 側では各文に重みをつけながら入力することで「要約文章としての文脈」を読み込んでいくことになる．

t 番目の文を要約に含むべき度合 p_t は以下のように計算される．

$$p_t = g(\bar{\boldsymbol{h}}_t : \boldsymbol{h}_t) \tag{3}$$

ここで g は分類器を表す．定性的には，すでに読み込まれた文脈情報 $\boldsymbol{h}_t$ と，要約文中の文脈情報 $\bar{\boldsymbol{h}}_t$ を用いて次の文の選出確率を計算することとなる．このようにしてすべての文のスコアが計算されたら，この中で上位数文が選出され，文が登場する順番に接続されて出力要約となる．

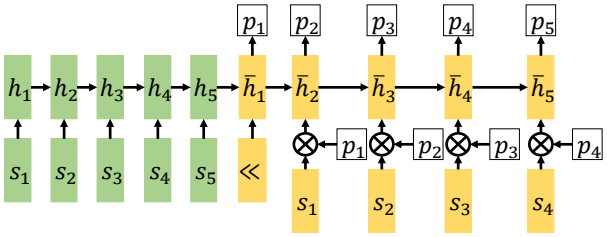

図 1: Cheng と Lapata のモデルの構造 [2]．図中の“≪”は終端記号を表すベクトルである．

3　提案モデル

Cheng と Lapata のモデルでは，全ての文を順次 RNN に入力して文書の読み込みを行っていた．しかし，論文のように文書が長くなると，RNN の系列長が長くなるので文脈の読み込みの精度が落ちると考えられる．さらに，学術論文の場合は各セクションごとに役割が異なり，あるセクションの末尾の文と次のセクションの頭の文でストーリーが続いているわけではないので，セクションを跨いだ文間で直接隠れ層状態を伝搬するのは適切ではない．

一方で，近年，RNN を階層構造や木構造に拡張するという研究が注目を集めている [3, 6, 7] が，文書の談話構造もまた階層構造であるといえる．論文の場合で言うと文書はセクションの並びで構成されており，各セクションは文の並びから構成されているからである．

そこで本研究では，Cheng と Lapata の手法を拡張して談話構造を活用した要約モデルを提案する．RNN を階層構造化するモデルは Li らの提案モデルを踏まえる [3]．対象とする文書が図 2 のような談話構造を持っていた場合を例にとり，モデルを図 3 に示す．

大まかな構造としては，文ベクトルの生成の段階までは Cheng と Lapata のモデルと同じで，文書の読み込みが異なる．基本的な考えは，全ての文を一気に読み込むのではなく, 各セクションごとに区切って読むということである．文の Encoder，Decoder およびセクションの Encoder，Decoder の四種類を用意する．各セクション内では，式 1 と同様に文の読み込みを行うが，セクション末の文のユニットの出力はセクションの Encoder に入力する．セクションの Encoder は所属している文を読み込み終わったら次のユニットに隠れ層状態を渡す．

$$\boldsymbol{v}_t = \mathrm{RNN}^{\mathrm{sec}}_{\mathrm{enc}}(\boldsymbol{h}_{\mathrm{end}}, \boldsymbol{v}_{t-1}) \tag{4}$$

この例では，まず INTRODUCTION 内の三つの文が

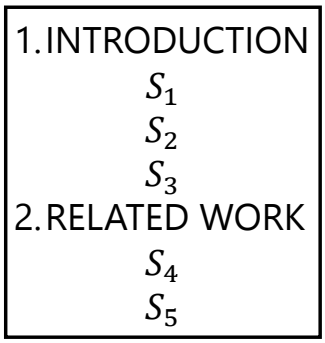

図 2: 学術論文の談話構造の例

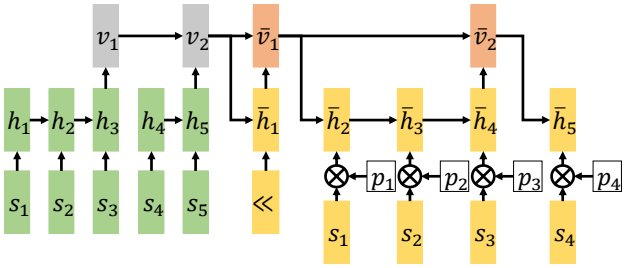

図 3: 提案モデルの構造

読み込まれた後，三つめの文の出力がセクションの Enocder の一番目のユニットに入力され，このユニットの出力が次のユニットに伝搬する．次に RELATED WORK 内二つの文も同様にして読み込まれ，この文書の読み込みは完了となる．

次に，セクションの Encoder の最後尾のユニットをセクションの Decoder の先頭に接続し，Decoder 側の処理に移る．セクションの Decoder 側でも同様にセクションごとに隠れ層状態を伝搬する．

$$\bar{\boldsymbol{v}}_t = \mathrm{RNN}^{\mathrm{sec}}_{\mathrm{dec}}(\bar{\boldsymbol{h}}_{\mathrm{end}}, \bar{\boldsymbol{v}}_{t-1}) \tag{5}$$

各セクション内で文を Decode する際には，セクション頭の文の RNN ユニットは一つ前のセクション Decoder から隠れ層状態を受け取る．

$$\bar{\boldsymbol{h}}_{\mathrm{top}} = \mathrm{RNN}^{\mathrm{sent}}_{\mathrm{dec}}(p_{\mathrm{top}}\boldsymbol{s}_{\mathrm{top}}, \bar{\boldsymbol{v}}_{\mathrm{prev}}) \tag{6}$$

この例では，Decoder 側ではまず INTRODUCTION 内の三つの文について重みをつけながら読み込むことで，このセクションの中で重要な文の情報を選択的に読み込んで，次のセクション内の処理に移る．

4　実験と結果

提案したモデルの性能を確認するため，学術論文をデータセットとした文抽出型の要約の実験を行った．

表 1: 各手法の ROUGE-1 および ROUGE-2 [%]

Models	ROUGE-1			ROUGE-2		
	Recall	Precision	F-score	Recall	Precision	F-score
OPTIMAL	65.2	68.4	66.7	30.5	32.1	31.2
LEAD	46.9	38.6	41.6	13.5	11.2	12.0
NN-SE	45.6	49.3	46.7	14.0	15.2	14.4
PROPOSED MODEL	45.0	46.4	45.0	13.4	13.9	13.4

4.1 データセットと評価手法

データセットとして，主要医学系雑誌等に掲載された論文のデータベースである PubMed Central[1]が提供している論文を用いる．論文データは xml 形式で提供されているが，各データからタイトル，キーワード，アブストラクト，本文を抽出した後，Stanford coreNLP[2]を用いて見出し語化および文分割をおこなう．実験には本文中に含まれる文の数が 300 以内，一文に含まれる単語の数が 200 以内の論文 30,000 本を用い，これらの 90%を学習データ，10%をテストデータとする．

評価については，テストデータ用の各論文から文抽出によって要約を作成し，その論文のアブストラクトと照らし合わせて ROUGE [4] で精度を評価する．出力する要約の制限長については, 一般に文数で制限する方法と文字数で制限する方法が存在するが, 本実験においては文数で制限する方法を採用する．

4.2 教師あり学習のための参照要約の作成

教師あり学習を行うために，まず最初に学習データとして用いる論文の本文中のすべての文を正例と負例にラベル付けする必要がある．本実験においては，本文中で考えられうる文集合の内，最もアブストラクトと類似している文集合，すなわちアブストラクトとの ROUGE-1 の F 値が最大となるような文集合を探索し，その文集合に属している文を正例とし，それ以外を全て負例とラベル付けする．しかし，本文中で考えられうる文集合の数は膨大であるから，動的計画法によって探索を行う [8]．

[1]ftp://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_bulk/
[2]http://stanfordnlp.github.io/CoreNLP/

4.3 モデルの設定

モデルの内部で用いる RNN はすべて Long Short-Term Memory（以下 LSTM）とし，式 3 における分類器 g はロジスティック回帰を用いた．

また，単語，文，文書の Embedding の次元はそれぞれ 100，200，400 とし，単語の Embedding に関しては word2vec[5] を学習データの論文に適用して得た Embedding 行列を初期値とする．

文の Embedding に関しては一文の中に含まれる単語について，bidirectional LSTM の出力の平均を用いることとし，各セクションの終端と文書の終端にはそれぞれ終端を表すベクトルを入力することとする．

ミニバッチサイズは 32 とし，確率的勾配降下法を用いて学習を行う．学習の過程では curriculum learning strategy という手法 [1] を適用する．Decoder 側のユニットへの入力には一つ前のユニットの出力 p_{t-1} を用いているため，特に予測ミスの起きがちな学習の初期段階では，p に関して間違えた値を入力することが連続してしまい学習が進みづらいと考えられる．この手法は，学習の序盤ではもし p の予測ミスが起きても，次のユニットへの入力の際は 4.2 節で教師信号として事前に得られている値 (負例ならば 0，正例ならば 1) を用いることとし，学習が進むにつれ徐々に入力の補助を弱めるというものである．補助の減衰のさせ方としては様々存在するが，本実験においては逆シグモイド関数の形で減衰していくモデルを適用した．

4.4 結果

結果を表 1 に示す．表中の OPTIMAL は 4.2 節で述べた手法により作成した参照文集合を表す．そのため，OPTIMAL のスコアは今回用いたデータセットにおいて，文抽出によって達成されうる理論上の上限値である．LEAD は文書の先頭から順に数文抽出する

手法，NN-SE は Cheng と Lapata の手法 [2] を我々が独自に実装したモデル，PROPOSED MODEL は本研究の提案モデルである．まず NN-SE および PROPOSED MODEL が LEAD を大きく上回っていることがわかる．その一方で，提案手法である PROPOSED MODEL は NN-SE を下回る結果となった．

4.5 考察

本実験において提案モデルがベースラインモデルを下回る精度となった原因としては，次の二点が考えられる．一つ目として，Decoder 側において各セクションの文頭の RNN ユニットは一つ前のセクション Decoder の隠れ層状態を受け取っているが，このようにするとかえってセクション中での文の位置の情報が損なわれてしまった可能性がある．セクション中の位置の情報を保持するための対策として，一つ前のセクション Decoder ユニットの出力を隠れ層状態として受け取るのではなく，入力として受け取ることが考えられる．また，二つ目として，本実験においてセクションの Embedding の次元は 400 であったが，これぐらいの大きさではセクションの内容の表現能力としては足りなかった可能性がある．

5 おわりに

本研究では，学術論文という比較的サイズの大きな文書に対して精度の良い要約を可能にするため，談話構造に着目した．教師あり学習に基づく文抽出器の手法の一つとして Encoder-Decoder モデルが提案されていたが，本稿ではその機構を踏まえてセクションの Encoder-Decoder を用いてより階層的に処理を行うモデルを提案した．今回の提案で導入したセクションの Encoder および Decoder は，RNN をセクション単位でまとめることによって文書読み込みの精度を上げるのが役割だったが，それだけでは直接的には要約の精度を向上させるには至らなかった．そのため，4.5 節で述べたように，Decoder 側において各セクション内の文の位置情報を保存すること，およびセクション単位での意味表現を可能にするためのパラメタ数を検討するのが次の課題となる．

今後の研究の方向性として，セクションレベルでのアテンションモデルを組み込んで，現在注目している Decoder のユニットにおいて入力文書のうちのどのセクションが関わりが最もあるかということを明確化し，より直接的にセクションの意味表現を参照することで精度向上を図ることが考えられる．他にも，今回の提案モデルにおいてはすべてのセクションを文書に登場する順に直列に接続したが，より本来の談話構造に忠実にして木構造型に RNN を接続したり，セクションの内部をより細分化して段落で区切ってみるのも面白いであろう．

参考文献

[1] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*. 2015.

[2] Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. In *ACL*, pp. 484–494, 2016.

[3] Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. A hierarchical neural autoencoder for paragraphs and documents. In *ACL-IJCNLP*, pp. 1106–1115, 2015.

[4] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL 2004 Workshop on Text Summarization Branches Out*, pp. 74–81, 2004.

[5] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pp. 3111–3119. 2013.

[6] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. Improved semantic representations from tree-structured long short-term memory networks. In *ACL*, pp. 1556–1566, 2015.

[7] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *NAACL-HLT*, pp. 1480–1489, 2016.

[8] 中須賀謙吾, 鶴岡慶雅. 談話構造を利用した学術論文の自動要約生成. 言語処理学会第 21 回年次大会発表論文集, pp. 569–572, 2015.